Text-To-Speech Infusion Into AWIPS Applications

(Design Presentation)

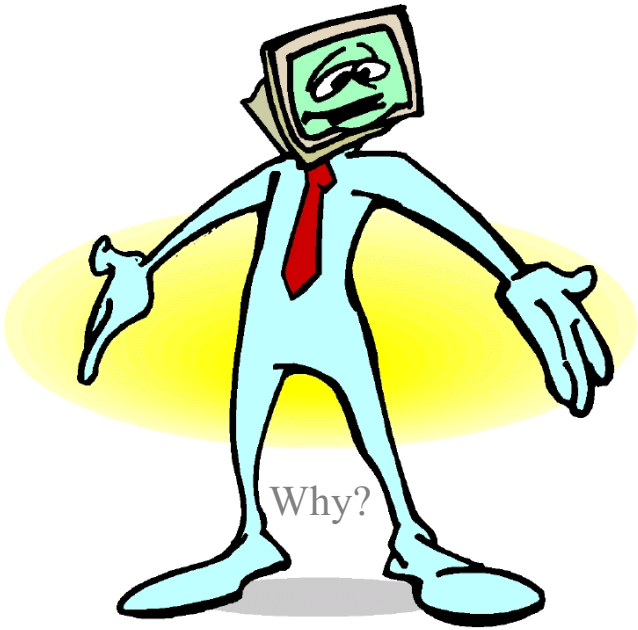
Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

November 5, 2002

Expectations / General Info / Assumptions

- Classify the usefulness
  - Proceeding with this not set in stone
  - Many have expressed an interest
  - Technologically doable
- WWA working group discussed
  - Encouraged to move forward on a design
- Linux audio driver fixed & speakers attached
- If pursued next step
  - Detailed design document (UML based)
    - Design improvements ideas welcome

By utilizing the NOAA Weather Radio Console Replacement System's Voice Improvement Processor (VIP), text-to-speech (TTS) capabilities can be introduced into AWIPS applications to serve a variety of operational needs.



Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Why?

(Quality Control - QC)

- Main: To provide a level of quality control difficult to address through error checking algorithms.
  - Specific target: product free text input
    - Overview & Synopsis
    - Descriptive Text
    - Eg: Winter Storm Warning products (WSW)
  - QC, audio playback of free text
    - Improve supplied text (grammar etc..)
  - QC, full product playback
  - QC, same TTS used for NWR broadcasts
    - Also promotes improved NWR broadcast quality
  - Closest RWP - RWP288 (QC)

Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Why?

(Other not so obvious uses)

- Positive Side Effect (Possibly?): By design, any AWIPS application will have limited access to VIP's TTS technology.
  - Audible alarms
  - Informational Announcements
    - Online training, cookbook instructions, etc...
  - Instructional Announcements
    - Install scripts, configuration interfaces, etc...
  - NWR Browser (view & hear text)
  - Others...?
- WWA-
  - None of the above
  - QC addressed only

Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Software Architecture

(General Info)

- Client / Server Model (TCP):
  - Server:
    - Application Server: *AppServer*
    - Standalone application
    - Run on VIPv3.0 or greater (linux 7.3)
    - Executed when VIP main interface is launched
      - Icon on turnkey system
      - Independent of VIPserver (start/stop)
    - Static entry port #7000
    - Maintains unique logging through log viewer interface
    - Speechify by Speechworks Inc (SWI)
      - TTS COTS application integrated into VIP
      - www.speechworks.com

Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Software Architecture

(General Info)

- Client / Server Model (TCP): (cont)
  - Server: (cont)
    - Proposed 7 port maximum simultaneous connections
      - 8 port license
      - 8 port system resource max
        - Speechworks benchmark
      - 1 port for intended VIP operation
      - 7 ports currently idle
    - *Fork* process utilized for multiple client connections
    - Four voice types supported
      - Craig - current (VIPv2.0.2)
      - Donna - current (VIPv2.0.2)
      - (Tom) - coming soon
      - Paulina - next build (VIPv3.0)

Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Software Architecture

(General Info)

- Client / Server Model: (cont)
  - Client:
    - Available to AWIPS applications through a shared object (.so)
      - .so files keep incorporating executables small
      - Realtime access
      - Prevents recompiling of every application as in the case of library usage (.a) IE...promotes a patchable process
    - Linux support only
    - Three major functions
      - Auto play
      - Audio player graphical user interface (GUI)
        - Audio functions provided (*play*, *pause*, *stop*)
      - Audio .wav/.mp3 file transfer
        - Audio functions provided/supported by participating application

Software Architecture

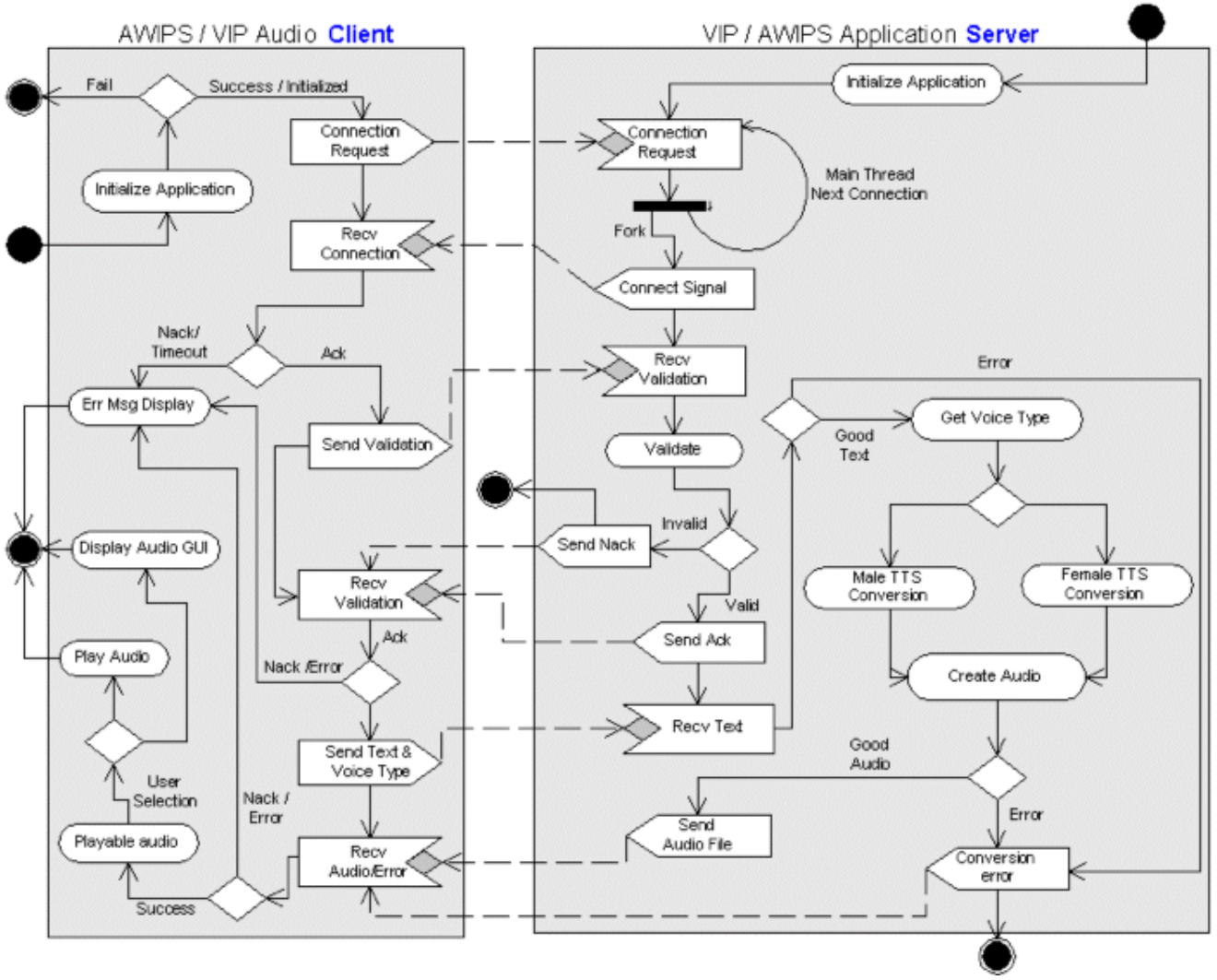
(General Info)

- Languages / COT software
  - VIP Application Server: C
  - AWIPS Client (.so): C & TCL/TK
    - Auto play
      - Sound eXchange (SoX) application
      - Is Linux standard
      - http://home.sprynet.com/~cbagwell/sox.html
    - Audio player GUI
      - Internal TCL/TK interpreter
        - Compiled into the, C based, shared object
      - Tcl.h & Tk.h
        - Not a Linux standard
      - Snack - http://www.speech.kth.se/SNACK/
        - Not a Linux standard

Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Software Architecture

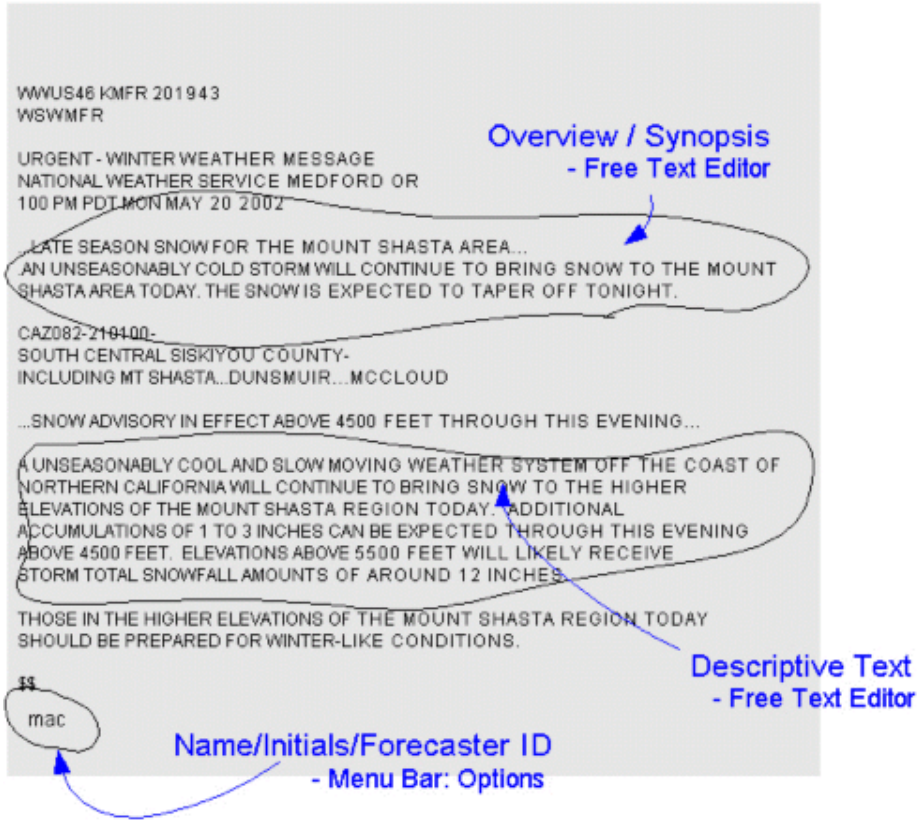
(UML: Activity Diagram)



Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

WWA Application Example

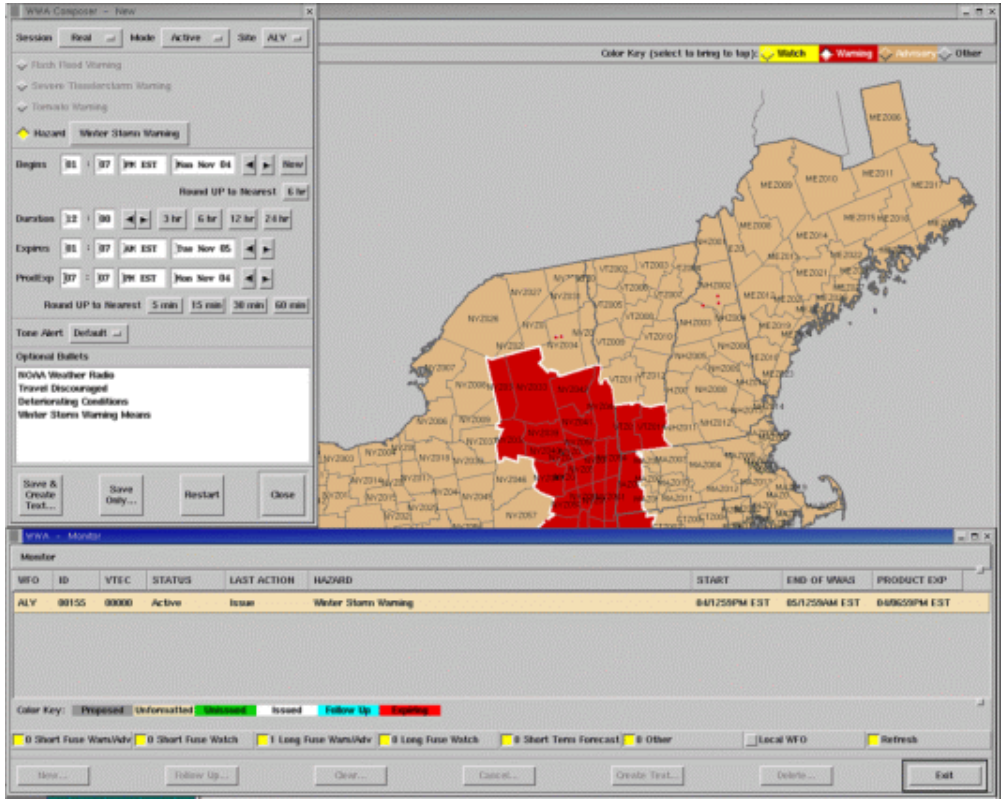
- WSW Example
  - WWA currently supplies each section of product
    - Except free text input
    - Image shows free text areas
  - Self-contained WWA (OB2)
    - Addresses capturing free text
    - Provides the text for TTS conversion



Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

WWA Application Example

- WWA 5.2.2 Image
  - 3 Interfaces
    - Monitor
    - Geo-Viewer
    - Composer
- Created products currently include
  - UGC
  - Time fields
  - Call to action statements
  - Headlines
  - Etc...

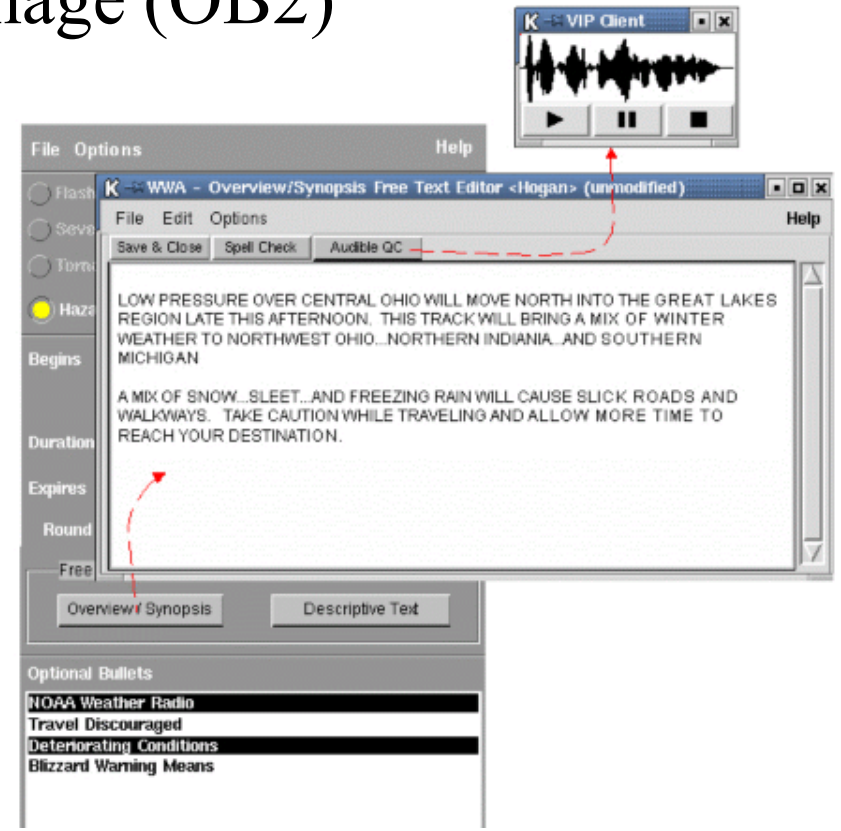


Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

WWA Application Example

(VIP Audio Client: Audio Player Function)

- Conceptual WWA Composer image (OB2)
  - Free text editor (FTE)
    - Provides capture text functionality
- Audible QC button added to FTE
  - Forecaster enters overview / synopsis and/or descriptive text
  - Prior to selecting *Save & Close* can perform an audible QC
    - Text sent to VIP Application Server for conversion
    - Audio player GUI displayed

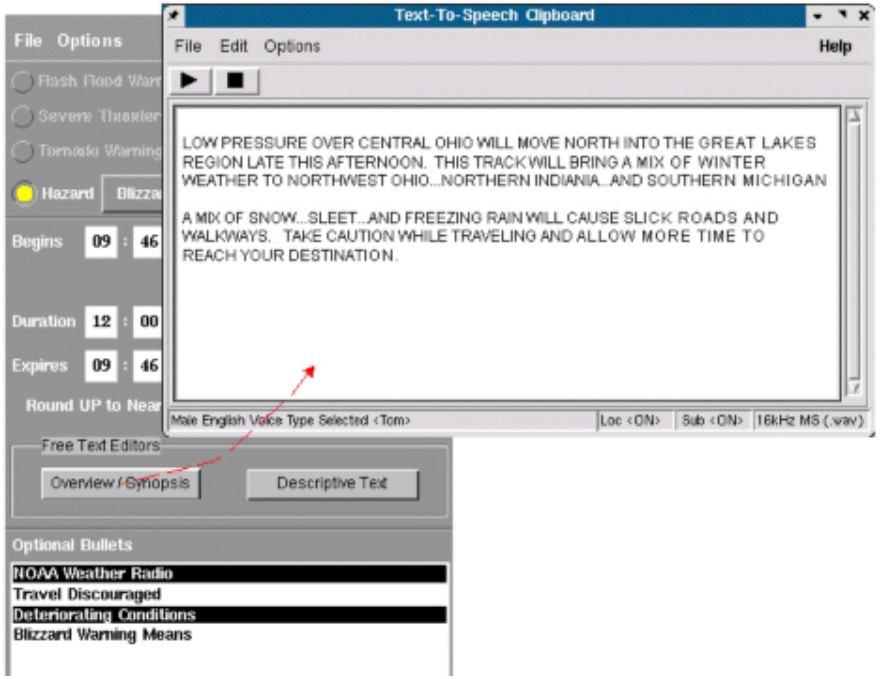


Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

WWA Application Example

(VIP Audio Client: .wav/mp3 Transfer)

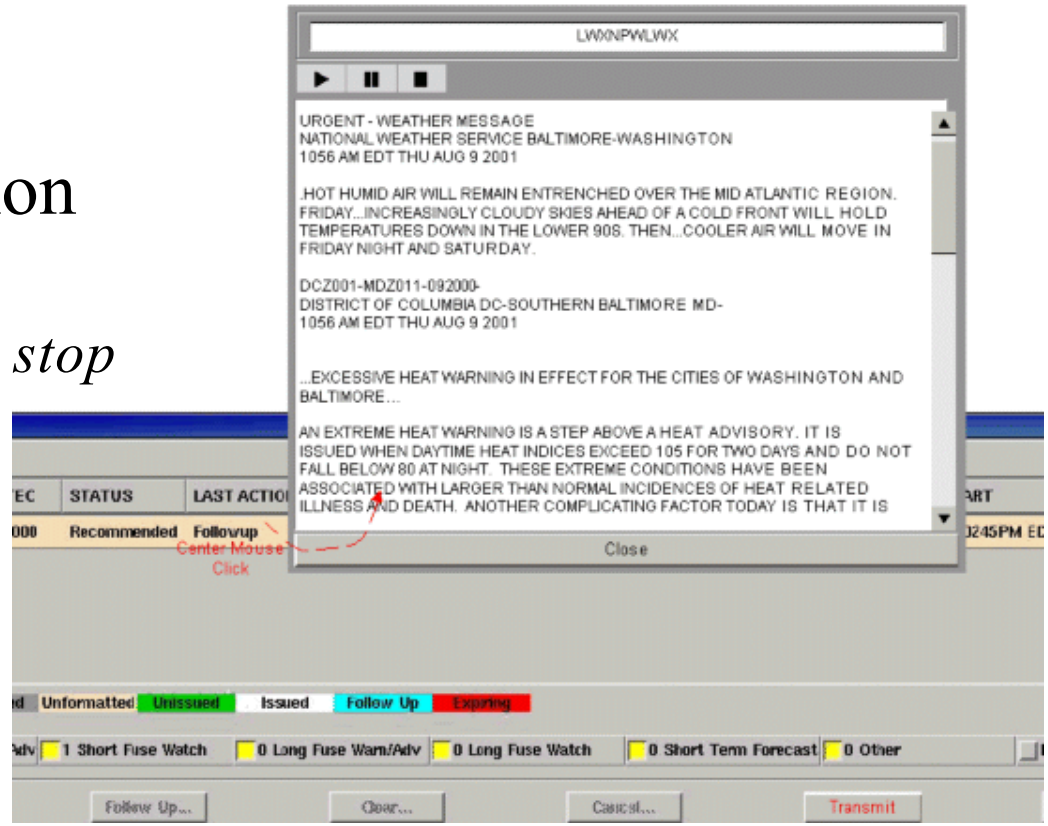
- Actual *play*, *pause*, *stop* buttons could be added FTE instead of an audible QC button
  - Forecaster enters overview/synopsis and/or descriptive text
  - Prior to selecting *Save & Close* the *Play* button is selected
    - Text sent to VIP Application Server .wav or .mp3 passed back
    - No GUI displayed audio functions managed by participating application



Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

WWA Application Example

- Conceptual WWA Self-contained Preview function
  - OBx
  - Could utilize *play*, *pause* & *stop* buttons
  - Could utilize audio playback GUI
  - Complete product read
    - Formatted text provided from current WWA NWR formatting process

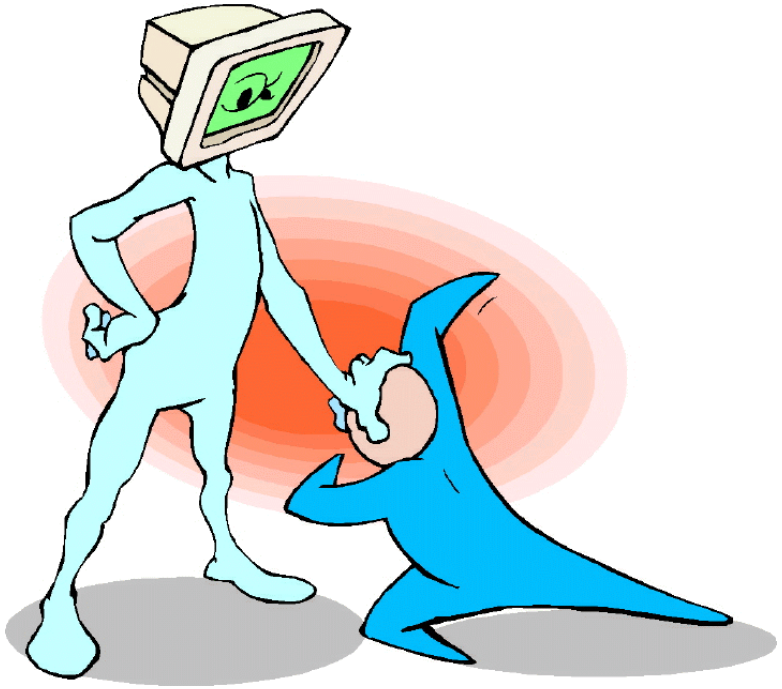


Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Security

(Not a critical issue for this application. However...)

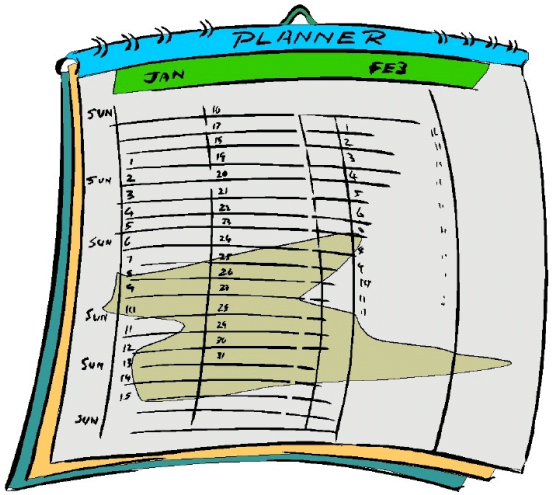
- AWIPS firewall security still applies to sockets
- Client / server connection verification
  - For registered users
- File permissions
  - awipsuser / group availability



Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Testing & Deployment Schedule

- Deployment
  - VIP
    - Application server included into VIPv3.0
    - Deployment begins May 5, 2003
  - AWIPS:
    - OB2 field testing / proof of concept
    - OB3 General Availability
- Testing
  - Benchmarking
    - VIP load (8 ports)
    - Network load (LAN)



Mark A. McInerney
National Weather Service
Meteorological Development Laboratory

Summary

- Flexible VIP audio client
  - Three user modes (playback, audio GUI, .wav/.mp3 transfer)
  - Shared object (.so)
- Limited VIP Application Server
  - 7 simultaneous VIP audio client connections
- WWA Proof of concept
  - OB2 field site testing
  - Deployment OB3
  - Application server deployed in VIPv3.0 (or later)
- WWA is my focus
  - Any other audio client use is left up to those that feel they have a need for TTS capabilities
- If pursued, next step is a detailed design (UML based)

Mark A. McInerney
National Weather Service
Meteorological Development Laboratory